Project STAR IST-2000-28764
Deliverable D6.3 Enhanced face and arm/hand detector

Date: August 29th, 2003

From: Katja Nummiaro, Rik Fransens and Luc Van Gool

Katholieke Universiteit Leuven, ESAT/VISICS,
Kasteelpark Arenberg 10, 3001 Heverlee, Belgium
Tel. +32-16-32.10.61 and Fax. +32-16-32.17.23
{knummiar, fransen, vangool}@esat.kuleuven.ac.be
http://www.esat.kuleuven.ac.be/ knummiar/star/star.html

To: STAR project partners
Project coordinator Artur Raczynski

Siemens CT PP6,
Otto-Hahn-Ring 6, 81730 Munich, Germany
Tel. +49-89-636.49.851, Fax. +49-89-636.481.00
artur.raczynski@mchp.siemens.de

1 Introduction

KU Leuven is responsible for the work package number 6, *Automated view selection and camera hand-over*. The main goal is to build an intelligent virtual editor that produces as an output a single video stream from multiple input streams. The selection should be made in such a way that the resulting stream is pleasant to watch and informative about what is going on in the scene. Face detection and object tracking is needed to select the best camera view from the multi-camera system.

KUL has delivered the STAR deliverables D6.1 *Initial face detection software* and D6.2 *Initial arm/hand tracking software* from work package 6, July 2002 (month 12). The integration of the detection and tracking has been needed to successfully provide this deliverable D6.3 *Enhanced face and arm/hand detector*.

We explain first the enhanced face detection, followed by the enhanced tracking software and finally the integration. Also the hand tracking results with simple histogram-based detection is presented. The results will be shown using the common STAR data sequencies, from different Siemens factories, in Germany.

2 Face Detection

The software by KUL can detect faces from single frames. The face detection software can determine
a) if there are faces in the image,
b) where these faces are located and
c) which size they have.

In STAR, the face detection has two main applications. The first one is to find initial regions for the tracker (see D6.2), i.e. the co-ordinates of the center of the face and the sizes of the half axes of the ellipse that surrounds the face. Faces are important for communication, so the second application uses face detection to decide which camera gives the best view. This deliverable explains only the use for the initialization of the tracker.

2.1 Method

We have implemented a Support Vector Machine (SVM) for face detection [1]. SVMs are large margin classifiers which use *Structural Risk Minimisation* to obtain the optimal hyperplane to separate the data. Kernels are used to allow for non-linear decision boundaries. The training of SVMs is supervised, i.e. during the training phase, manually labeled training data are fed into the optimisation routine, and the goal is that the obtained classification function generalizes well over unseen examples.

The classification algorithm locates the faces in the image by sliding a window of fixed size over the image and classifying the image content within the window as either a face or a non-face. The size of the window is fixed (in our current implementation 32x32 pixels), and in order to find faces of different sizes the procedure is applied to a pyramid of successively downscaled versions of the input image. The operation of the classifier can be subdivided into 6 steps. The first two are quick rejection procedures based on the variance and fraction of skin color of the image window. If either the variance is too low (i.e. the window is part of a homogenous intensity region) or the fraction of skin color is too low, the window is immediately discarded. Both steps are computed using integral images, a technique which allows for efficient computation of rectangular features (e.g. the sum of pixel values, the average value, the variance,...) by exploiting redundancies in the computation of nearby regions. The third step is a statistical normalisation procedure, which aims at robustness against global lighting conditions [2]. Next, the window is projected onto a subspace to reduce the dimensionality of the classification problem. This allows faster computation of the decision as fewer features are taken into account. The basis of the subspace is obtained by searching for those directions which are most used by the SVM classifier to discriminate between faces and non-faces in the original space. The fifth and sixth step involve the actual SVM-classification. Both classifiers are the result of the *Reduced Set* method to compress classically trained SVMs, where the first classifier is compressed with loss of accuracy (100 SVs retained) and the second classifier is compressed without loss of accuracy (490 SVs retained). Both classifiers operate in a cascade, with a conservative threshold on the first one, to ensure that no faces are lost in this step.

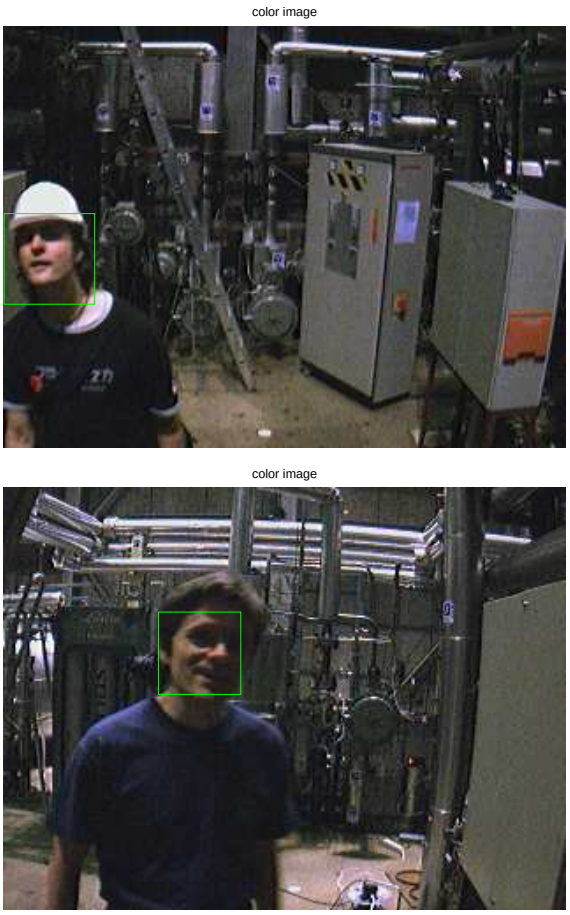


Figure 1: Some results of the face detection software.

Obviously, the global classification time is a function of the image characteristics, and depends heavily on how many frames can be discarded from further computation by the quick rejection units. On some typical examples, see for example Figure 1, the running time was in the order of seconds.

2.2 Results

Figure 1 shows some successfull detections. However, we also encountered some difficulties with certain fragments of the test videos, due to the following reasons:
1) the face is not in a frontal view,
2) the face is too small (i.e. smaller than 32x32),
3) extreme lighting conditions (e.g. strong cast shadows from the head wear).

3 Tracking

The two-dimensional object tracking is needed for selecting the camera with the current best view. The object of interest (person, face, hand, arm) can be tracked with all the cameras simultaneously or by using only one camera at the time. This requires information exchanges between the cameras and successful initialization and re-initialization of the tracked object. Using multiple cameras with known calibration information, do anywhow make the tracking more robust and intelligent, as the information about the object is shared and basic constraints like epipolar geometry can be used.

This deliverable concentrates on single-camera tracking, together with the face detection as an initialization step for the tracker. In the final deliverable D6.4 *Camera Hand-Over Package* we deliver the multi-camera based tracking, together with the best-view selection.

We have enhanced the software of deliverable D6.2 – the software is based on particle filtering [3] and the similarity measurements between the target model histogram and the propagated candidate histograms [4].

3.1 Method

The used tracking method is based on particle filtering, together with color-based image features that characterize the tracker object. The key idea of particle filtering is to apply a recursive Bayesian filter based on weighted sample sets. This probabilistic approach provides a robust tracking framework, as it models uncertainty, can keep its options open and considers multiple state hypotheses simultaneously. The integration of color distributions into particle filtering has many advantages for tracking non-rigid objects. Color histograms in particular are robust to partial occlusion, are rotation and scale invariant and are calculated efficiently.

The target model of the particle filter is defined by the color information of the tracked object. As the tracker should find the most probable sample distribution, the target model is compared with the current hypotheses of the particle filter using the Bhattacharyya coefficient, which is a popular similarity measure between two distributions. As the color of an object can vary over time dependent on the illumination, visual angle and camera parameters, the target model is adapted during temporally stable image observations.

The proposed color-based particle filter [4] belongs to the top-down approaches as the image content is only evaluated at the hypothetical object positions. In section *Publications* all the publications that cover the topic and have been published during the STAR project by KUL, are listed. The details of our tracking algorithms and some result sequencies are shown in those publications, but also on the KUL-STAR webpage http://www.esat.kuleuven.ac.be/~knummiar/star/star.html.

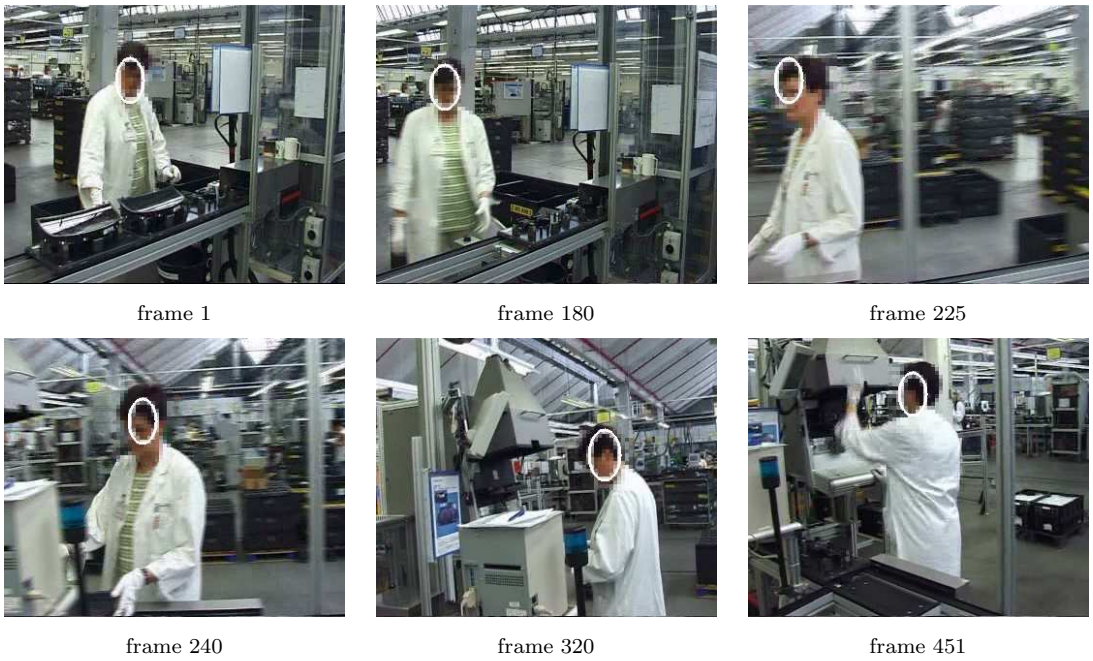


Figure 2: The *factory* sequence shows the tracking performance in Siemens factory, Germany.

3.2 Results

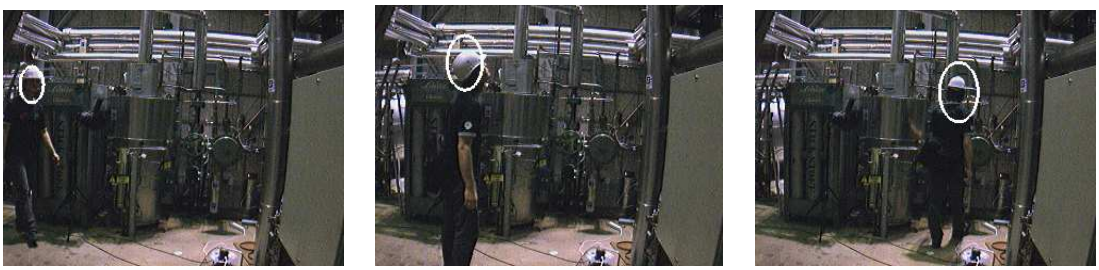
The implementation of the tracker is done with C++. We use a 800 Mhz Pentium-3 PC that is running under Linux Redhat7.3 and that is connected to a SONY DFW-VL500 digital camera (image size of 640x480 or 320x240 pixels). The use of sequencies of arbitrary sizes is also possible.

The capturing software is able to grab 15-30 frames per second with the SONY digital camera. These frames are used as input for the tracker. The use of RGB color space with 8x8x8 bins has been found to be the most efficient and robust for the color-based particle filter. The target model region has the shape of an ellipse.

Figure 2 illustrates the tracking results of the industrial application, in the Siemens factory. This factory is used for the 1st scenario of the STAR project.The tracked person carries out several operations on different machines, resulting in large out-of-plane rotations of the head and in unpredictable locations in front of a highly textured background. The color-based particle filter handles all requirements successfully over this 450 frame sequence with only 100 samples. For reasons of anonymity, the face of the tracked person in the result images is pixelized, but the tracker was running on the original images.

The computing time to process one frame depends mainly on the region size of the object, the number of samples in the tracking method and the image size. The tracking is running in real-time (15-30 fps) without any special optimization.

Figures 3 and 4 show the tracking results in another industrial environment and application. These sequences were taken in the Framatome factory in Erlangen, for the common STAR



frame 40 frame 60 frame 80

frame 100 frame 120 frame 130

Figure 3: The *worker A* sequence shows the single camera tracking performance in the Framatome factory, Germany.

frame 40 frame 60 frame 80

frame 100 frame 120 frame 130

Figure 4: The *worker B* sequence shows the single camera tracking performance in the Framatome factory, Germany.

scenario. The capturing was synchronized, thus the frame numbers correspond to the same time of the sequence.

Figure 5 illustrates the use of the same kind of tracker applied to a hand rather than the face of a worker. As a dedicated hand detector is more difficult to develop, we rather look for regions that have colours that are in agreement with a learned hand colour histogram. The

Figure 5: The tracker is capable of following hands of the worker.

region size is restricted to an expected range of sizes [4]. As the figure shows, the tracker is not very precise in terms of the hand position (and was not intended bo be), but exchanges precision for robustness. The shown sequence features hand occlusions, strong changes in relative viewpoint and hand configuration, and substantial changes in scale.

4 Integration of Face Detection and Tracking

For the integration we can run the face detection algorithm to get the starting image coordinates and scale factor (Hx and Hy of the initial ellipse) for the face target, and start the tracking immediately when the face is found.

5 Conclusion

The overview of the methods and results of deliverable D6.3 *Enhanced face and arm/hand detector* from workpackage 6 of STAR IST-2000-28764 project has been presented in this report. The face detection algorithm can detect faces from single images and the tracking algorithm is able to follow a known object (like a face) over a recorded image sequence in real-time, with a camera connected to a laptop/computer. The two algorithms can be run independently or used as an integrated software by initializating the tracker with the face detection. In cases of lost objects (faces), the face detection can be activated, and the tracker re-initialized the same way. In case af hands/arm, the detection is done by using the known color distributions of the hands of the worker by finding the region from the single images. The integrated version will be used in the STAR work package 6.4 *Camera hand-over* for choosing the best view camera stream from a multi-camera system. The application of the camera-hand-over will decide how the hands respect to the faces, ie. which rule is applied when choosing the best-view. The main issue will be deciding inside STAR collaboration if the final scenario concentrates on faces or hands or some specific combination of them.

Publications

The list of publications that cover the topic of STAR deliverable D6.3, done by KUL:

K.Nummiaro, E.Koller-Meier, L.Van Gool:
A Color-Based Particle Filter,
Proceedings of the 1st International Workshop on Generative-Model-Based Vision,
in conjunction with ECCV02, Denmark, pp. 53-60, Jun 2002.

K.Nummiaro, E.Koller-Meier, L. Van Gool:
Object Tracking with an Adaptive Color-Based Particle Filter,
Proceedings of the Symposium for Pattern Recognition of the DAGM,
LNCS 2449, Switzerland, pp. 353-360, Sep 2002.

K.Nummiaro, E.Koller-Meier, L.Van Gool:
Color-based Real-time Recognition and Tracking,
Abstract and demonstration in the IEEE Int. Symposium in Mixed and Augmented Reality
ISMAR2002, CDROM Proceedings, Germany, Oct 2002.

I. Geys, L.Van Gool:
Virtual Label Extraction and Tracking,
Abstract and demostration in the IEEE Workshop on Applications of Computer Vision 2002,
USA, Dec 2002.

K.Nummiaro, E.Koller-Meier, L.Van Gool:
An Adaptive Color-Based Particle Filter,
Image and Vision Computing, Vol. 21, Issue 1, pp. 99-110, Jan 2003.

K.Nummiaro, E.Koller-Meier, L.Van Gool:
Color Features for Tracking Non-rigid Objects,
Chinese Journal of Automation, Vol 29, No. 3, pp.345-355, May 2003.

K.Nummiaro, E.Koller-Meier, T.Svoboda, D.Roth, L.Van Gool:
Color-Based Object Tracking in Multi-Camera Environments,
Symposium for Pattern Recognition of the DAGM, Germany, Sep 2003.

References

[1] C.J.C. Burges, *A tutorial on support vector machines for pattern recognition*, Data Mining and Knowledge Discovery, Vol.2(2): pp. 955-974, 1998.

[2] A.S. Georghiades, P.N. Belhumeur and D.J. Kriegman, *From Few to Many: Illumination Cone Models for Face Recognition Under Variable Lighting and Pose*, Pattern Analysis and Machine Intelligence Vol.23, No.6, 2001.

[3] M. Isard and A.Blake, *CONDENSATION - Conditional Density Propagation for Visual Tracking*, International Journal on Computer Vision, 29(1): pp. 5-28, 1998.

[4] K. Nummiaro, E. Koller-Meier and L. Van Gool, *An Adaptive Color-Based Particle Filter*, International Journal of Image and Vision Computing, pp. 99-110, Vol 21(1), 2003.